

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平 7 - 2 8 5 7 9

(43) 公開日 平成 7 年 (1995) 1 月 31 日

(51) Int. Cl. ⁶

G 0 6 F 3/02

3/03

識別記号

3 9 0 Z

3 8 0 M 7165-5 B

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数 1 3

O L

(全 1 8 頁)

(21) 出願番号 特願平 5-169411

(22) 出願日 平成 5 年 (1993) 7 月 8 日

(71) 出願人 000002369

セイコーエプソン株式会社

東京都新宿区西新宿 2 丁目 4 番 1 号

(72) 発明者 久松 豊

長野県諏訪市大和 3 丁目 3 番 5 号セイコーエ
プソン株式会社内

(72) 発明者 小池 袈裟光

長野県諏訪市大和 3 丁目 3 番 5 号セイコーエ
プソン株式会社内

(72) 発明者 青木 永亘

長野県諏訪市大和 3 丁目 3 番 5 号セイコーエ
プソン株式会社内

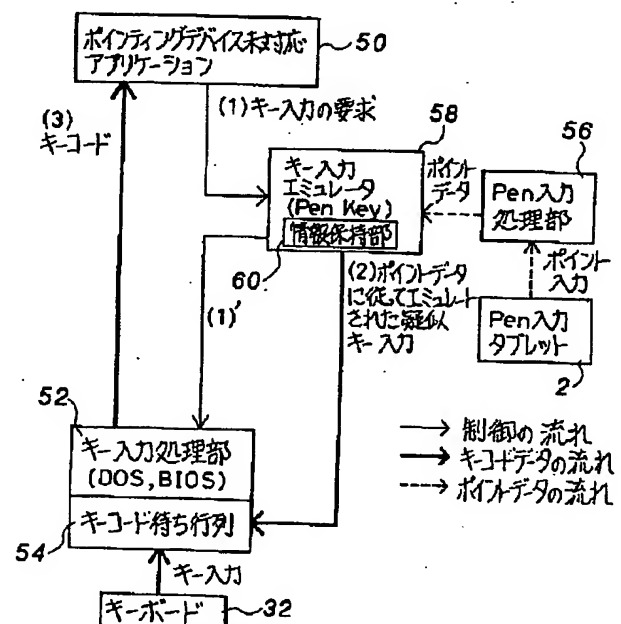
(74) 代理人 弁理士 鈴木 喜三郎 (外 1 名)

(54) 【発明の名称】 情報処理装置

(57) 【要約】

【目的】 ポインティングデバイスに対応していないアプリケーションプログラムであっても、ポインティングデバイスによる軽快な操作が可能な情報処理装置を提供する。

【構成】 アプリケーション 50 が割り込みでキー入力の要求をすると、Pen key 58 が起動し、Pen 入力タブレット 2 から入力されたポイントデータがあれば、それによって所定のキーコードを生成して、あたかもキーボード 32 から入力されたかのように、キーコード待ち行列 54 にそのキーコードを設定する。キー入力処理部 52 ではキーコード待ち行列 54 から順々にキーコードを取り出すので、設定された擬似キー入力のデータが、アプリケーション 50 に返される。



本発明の全体の流れを示す図

1

【特許請求の範囲】

【請求項1】 ポインティングデバイス等による位置情報を保持する位置指定手段と、
前記位置指定手段により出力された情報を、キーコード情報に変換する変換手段と、
前記キーコード情報を保持するコード処理手段と、
前記コード処理手段からの情報で制御される表示手段と、
を有する情報処理装置。

【請求項2】 請求項1記載の情報処理装置であって、
前記変換手段は、現在のカーソル等の位置表示手段を、現在のポインティングデバイス等により指定する位置に移動するよう、キーコード情報を出力することを特徴とする情報処理装置。

【請求項3】 請求項1記載の情報処理装置であって、
前記キーコード情報は、特定の待ち時間において出力されることを特徴とする情報処理装置。

【請求項4】 請求項3記載の情報処理装置であって、
前記待ち時間は、入力モードに合わせて可変とすることを特徴とする情報処理装置。

【請求項5】 ポインティングデバイス等による位置情報を保持する位置指定手段と、
前記位置指定手段により出力された情報を、キーコード情報に変換する変換手段と、
前記キーコード情報を保持するコード処理手段と、
前記コード処理手段からの情報で制御される表示手段と、
を有する情報処理装置であって、

前記変換手段は、現在のカーソル等の位置表示手段を、現在のポインティングデバイス等により指定する領域に基づいて、前記位置表示手段を移動するよう、キーコード情報を出力することを特徴とする情報処理装置。

【請求項6】 請求項3記載の情報処理装置であって、
前記領域は境界線を略X字型に設けていることを特徴とする情報処理装置。

【請求項7】 請求項3記載の情報処理装置であって、
前記領域は境界線を略H字型に設けていることを特徴とする情報処理装置。

【請求項8】 請求項1記載の情報処理装置であって前記変換手段は複数個であり、動的に切替ることを特徴とする情報処理装置。

【請求項9】 ポインティングデバイス等による位置情報を保持する位置指定手段と、
前記位置指定手段により出力された情報を、キーコード情報に変換する変換手段と、
前記キーコード情報を保持するコード処理手段と、
前記コード処理手段からの情報で制御される表示手段と、
を有する情報処理装置であって、
前記表示手段の特定の領域を位置指定すると、対応して

2

設けられたキーコードあるいはシフトコードを出力することを特徴とする情報処理装置。

【請求項10】 請求項9記載の情報処理装置であって、
前記特定の領域は前記表示画面の最下部付近であることを特徴とする情報処理装置。

【請求項11】 ポインティングデバイス等による位置情報を保持する位置指定手段と、
前記位置指定手段により出力された情報を、キーコード情報に変換する変換手段と、
前記キーコード情報を保持するコード処理手段と、
前記コード処理手段からの情報で制御される表示手段と、
を有する情報処理装置であって、

前記表示手段の特定の領域を位置指定する時に、特定の時間内に複数の位置を指定し、前記複数の位置に対応して設けられたキーコードあるいはシフトコードを出力することを特徴とする情報処理装置。

【請求項12】 請求項11記載の情報処理装置であって、
前記特定の時間内に複数の位置を指定する手段として、
ポインティングデバイスのスライド動作を使用することを特徴とする情報処理装置。

【請求項13】 請求項9記載または請求項11記載の情報処理装置であって、
前記シフトコードが出力され、シフト状態となった場合、
前記シフト状態を示すガイダンスを前記表示手段に1つまたは複数表示することを特徴とする情報処理装置。

【発明の詳細な説明】

【産業上の利用分野】 本発明は、ポインティングデバイス入力に対し未対応のアプリケーションプログラムを、
ポインティングデバイスを用いて操作可能とした情報処理装置に関する。

【従来の技術】 マウス、Pen等のポインティングデバイスをサポートするアプリケーションプログラム（以下：対応アプリケーション）は、マルチウィンドウOSに対応するものを主として増えつつあるが、比較的古いOSであるDOS等上では、いまだポインティングデバイスに未対応のアプリケーション（以下：未対応アプリケーション）が多く存在し、しかも主流として使用されている。しかし、携帯性を重視した情報機器では、入力装置はポインティングデバイスのみとして、キーボード等のキー入力デバイスを持たないものもあり、このような情報機器上においても未対応アプリケーションを動作させる必要性が生じている。図20は対応アプリケーションの動作を示す図である。ポインティングデバイスを用いてアプリケーション上でカーソル200を移動させる場合、ポイント指示カーソル202の移動先の持っていない、クリック等を行えば、専用のドライバから位置情報あるいは移動量情報としてアプリケーションに渡され、アプリケーションがカーソル200を移動する。しかし未対応アプリケーションは位置情報あるいは移動量情報

を受け取る手段を有しておらず、カーソル200を移動するにはキー入力部から「右向き矢印」「下向き矢印」等のキーコードを受け取るしか方法がない。

【発明が解決しようとする課題】ポインティングデバイス入力においても、図21に示すように、ソフトウェアキーボード204を表示し、希望のキーをクリックすることにより、対応するキーコードを発生させることも考えられているが、あくまでも回避手段であり、操作の一環性が失われたり、ソフトウェアキーボード204によってカーソル200の移動先の位置P2が見えなくなるなど、操作性を極端に低下させるという問題点があった。本発明は、このような問題を解決し、未対応アプリケーションであっても、ポインティングデバイスによる軽快な操作が可能な情報処理装置を提供することを目的とする。

【課題を解決するための手段】本発明の情報処理装置は、図1に例示するように、ポインティングデバイス等による位置情報を保持する位置指定手段M1と、前記位置指定手段により出力された情報を、キーコード情報に変換する変換手段M2と、前記キーコード情報を保持するコード処理手段M3と、前記コード処理手段からの情報で制御される表示手段M4と、を有することを特徴とする。また前記変換手段は、現在のカーソル等の位置表示手段を、現在のポインティングデバイス等により指定する位置に移動するよう、キーコード情報を出力することを特徴とする。また、前記キーコード情報は、特定の待ち時間において出力されることを特徴とする。また、前記待ち時間は、入力モードに合わせて可変とすることを特徴とする情報処理装置。また、ポインティングデバイス等による位置情報を保持する位置指定手段M1と、前記位置指定手段により出力された情報を、キーコード情報に変換する変換手段M2と、前記キーコード情報を保持するコード処理手段M3と、前記コード処理手段からの情報で制御される表示手段M4とを有する情報処理装置であって、前記変換手段は、現在のカーソル等の位置表示手段を、現在のポインティングデバイス等により指定する領域に基づいて、前記位置表示手段を移動するよう、キーコード情報を出力することを特徴とする。また、前記領域は境界線を略X字型に設けていることを特徴とする。また、前記領域は境界線を略H字型に設けていることを特徴とする。また、前記変換手段は複数個であり、動的に切替えることを特徴とする。また、ポインティングデバイス等による位置情報を保持する位置指定手段M1と、前記位置指定手段により出力された情報を、キーコード情報に変換する変換手段M2と、前記キーコード情報を保持するコード処理手段M3と、前記コード処理手段からの情報で制御される表示手段M4とを有する情報処理装置であって、前記表示手段の特定の領域を位置指定すると、対応して設けられたキーコードあるいはシフトコードを出力することを特徴とする。また、前

記特定の領域は前記表示画面の最下部付近であることを特徴とする。また、ポインティングデバイス等による位置情報を保持する位置指定手段M1と、前記位置指定手段により出力された情報を、キーコード情報に変換する変換手段M2と、前記キーコード情報を保持するコード処理手段M3と、前記コード処理手段からの情報で制御される表示手段M4とを有する情報処理装置であって、前記表示手段の特定の領域を位置指定する時に、特定の時間内に複数の位置を指定し、前記複数の位置に対応して設けられたキーコードあるいはシフトコードを出力することを特徴とする。また、前記特定の時間内に複数の位置を指定する手段として、ポインティングデバイスのスライド動作を使用することを特徴とする。また、前記シフトコードが出力され、シフト状態となった場合、前記シフト状態を示すガイダンスを、前記表示手段に1つまたは複数表示することを特徴とする。

【実施例】以下に、本願記載の情報処理装置の好適な実施例について説明する。図2は一実施例としての情報処理装置16の平面図(a)、及び左の側面図(b)、右の側面図(c)である。情報処理装置16は、上面にLCDパネルからなる表示部4を備える。その背面には電磁誘導方式のPen入力タブレット2が備えられており、Pen6を近づけることにより、その先端の情報を情報処理装置16に入力することができる。後端にはバッテリー8が配置されている。左側面(b)にはカバー10が設けられ、内部にはICカード、HDD等の外部記憶装置、内蔵ファックスモデムカード等が接続可能なコネクタがある。右側面(c)のカバー12、カバー14の内部にはそれぞれICカード、外部FDDを接続するためのコネクタが用意されている。また、図示しないが、情報処理装置16の底面にはキーボード接続用のコネクタが設けられている。この情報処理装置16は小型軽量で可搬性に富むように設計されており、キーボードI/Fを備えてはいるが、通常はPen6のみですべての操作を行えるようになっている。図3は本発明の情報処理装置のハードウェア構成図である。CPU20はROM22内の基本制御プログラム(以下BIOS)により装置全体を制御している。システムバス40には他に、主記憶を構成し、OSやアプリケーションプログラム、各種制御プログラム、後述する情報保持部60を格納するRAM24、Pen入力タブレット2が接続され、XY座標を出力するタブレットI/F26、表示部4を制御する表示制御部28、ICカード、HDD、FDD等の外部記憶装置36が接続される外部記憶I/F34、タイマ38が接続されている。図4は本発明の全体の流れを示す図である。以下に動作を説明する。

- (1) アプリケーション50がINT21h、INT18h等の割り込みでキー入力の要求をすると、キー入力エミュレータ(以降Penkey)58が起動し
- (2) Pen入力タブレット2から入力されたポイント

データがあれば、それに従って所定のキーコードを生成して、あたかもキーボードから入力されたかのように、キーコード待ち行列54（通常のキーバッファと同義）にそのキーコードを設定する。制御は（1）'として、元々のキー入力処理部52に移行する

（3）キー入力処理部52ではキーコード待ち行列54から順々にキーコードを取り出すので、（2）で設定された擬似キー入力のデータが、未対応アプリケーション50に返される

情報保持部60は、Penkey58が参照するモードフラグ、変数等の格納場所として使われる。また本実施例においてPenkey58は主記憶上に常駐するプログラムあるいはそれに相当する機能を有するプログラムとして提供されるので、常にアプリケーションのキー要求を監視することができる。Penkey58の提供する機能は、以下の通りである。

（1）カーソル移動機能

カーソル移動機能は、Penでポイントした位置に応じて、カーソル移動キーのキーコードを適宜出力することによって実現する。これによって、Penによる操作で、未対応アプリケーションにおいても、カーソルの移動を行うことができる。カーソル移動モードは

- ・P（Pointing）モード
- ・X（X-divided）モード

の2つがある。Pモードは図12のように、移動先P2を直接ポイントし、その位置にカーソル移動させるモードである。Xモードは図13（b）のようにタブレットをX型に4領域に分け、それぞれに移動キーを割り当てるモードである。両モードは、図13（a）の表示画面のカーソル移動モードブロック72をPen6でポイントすることにより切り替えることができ、“P”、“X”のモード表示がされる。また、入力状況に応じた自動切替も可能である。

（2）ファンクションガイドエミュレーション機能

図17のように、アプリケーションは通常、表示画面の最下行にファンクションキーガイドを表示する。このファンクションキーガイドに重なるようにキーブロック、K1__76からKA__85を配置し、ガイド部をPen6でポイントすると、そのファンクションキーに対応したキーコードを送出する。このキーコードは、通常のキーボードと同様に、その時のシフト状態（CTRL、SHIFT、GRAPH等により指定）に応じて変化する。LB（レフトブランク74またはRB（ライトブランク）90をポイントするとシフト状態が通常（表示なし）～SHIFT（表示“S”）～SHIFT固定（表示“SL”）～通常（表示なし）と、トグルで変化する。また、ファンクションキーに限らず、[ESC]キー、[リターン]キーを配置したり、図18、図19のように、Pen6のポイントの仕方を変えることにより、より多くの機能を割り当てることができる。各動作モード

をさらに詳しく説明する

[Pモード詳細] このモードでは、Penでポイントした位置まで、自動的にカーソルが移動する。これを実現するために、カーソルが目的の位置に最も近づいたか、カーソルの移動が検出されなくなるまで、適切なキーコードが送出され続ける。スクリーンエディタ等のように、画面上でカーソルが自由に動くような状態において、最も効果的に動作する。どのようなカーソル移動キーコードが出力されるかは、カーソルの位置と、Penでポイントした位置との相対関係によって決まる。このモードにおいては、カーソル位置を検出してその動作を決定するが、PモードをXモード自動可変動作に設定しておく、Pモードの動作に対して、カーソルの移動が検出されなかった場合は、Pモードの動作を取り消して、Xモードに一時的に移行し、Penが離されるまで、Xモードの動作を行なわせることができる。動作は下記の通りである。

1) カーソル位置と、ポイントされた位置との画面縦方向のずれを0にするようにカーソル移動キーコードを送出する

2) 縦方向のずれが0になるまで、1)を繰り返す。縦方向のずれが0になった時点で、3)に移る縦方向のずれが0でないのに、カーソルの移動が認められない場合は、Pモードの動作の設定により、下記の動作に従う
・Xモード自動可変動作の場合、前回の動作を取り消し、Penが押しつけられている間だけ、Xモードの動作を行なう。動作終了後は、自動的にPモードに戻る（別途図を用いて説明する）

・Pモード固定動作の場合は、総ての動作を直ちに停止する。

3) カーソル位置と、ポイントされた位置との画面横方向のずれを0にするようにカーソル移動キーコードを送出する

4) 横方向のずれが最小になるまで、3)を繰り返す。下記に該当する場合は、オーバーランと判断して、直前の動作をキャンセルしてから停止する。（直前と逆に移動して、1つ前に戻る）

・カーソルが目標位置の左側から右側になった場合（全角文字の右半分に対応）

・カーソルの行位置が変化したとき（行末等での行の変化に対応…行の改行マークより右側をポイントしたとき、意図に反して次行にカーソルが飛ぶ場合を想定）

・距離の増加が7を越えたとき（TABコードの部分は一度の移動キーで通常8文字分移動する。これもポイントした位置に反する移動となる）

上記以外で、横方向のずれが0になったら、停止する。横方向のずれが0でないのに、カーソルの移動が認められない場合は、総ての動作を直ちに停止する。上記の通り、PモードがXモード自動可変動作に設定されていると、カーソルがカーソル移動キーにより正常な位置に移

動できない場合は、Pモードは自動的にXモードに切り替わる。カーソル移動キーを送出するタイミングは、キー入力待ち割込みを横取りして獲得する。但し、その割込みのたびにキーコードを送出すると、カーソル移動の実処理が間に合わなくなり、結果としてカーソルの移動が検出されず、Pモードの動作がすぐに停止してしまう恐れがあるので、これを避けるために、キーコードの送出は、キー入力待ちの割込みのタイミングを適当に間引いて（ウェイトを入れて）行なう必要がある。Penを押しつけ続けると、キーボードと同様のリピート機能が働く。このときの動作も、前記の原理に基づくため、Penを押しつけている位置までカーソルが到達すると、カーソルの動作は停止する。しかし、カーソルの移動する範囲が限られているとき、その範囲の外でPenを押しつけ続けると、リピート機能が働き続けるため、エディタ等でスクロール範囲の外（最上段のガイド部分等）でPenを押しつけ続けることにより、連続的にスクロールを行なうことができる。リピート機能の時間間隔（繰り返し開始までの時間、繰り返す周期）は、キーボードのリピート機能に近く設定するが、必要に応じて変更できる。

〔Xモード詳細〕このモードでは、図13（b）のように画面をX型にA、B、C、Dに4分割し、Penでポイントした位置に応じて、上／下／左／右のいずれかにカーソルが移動する。例えば、画面右側中央部をポイントすると、右側にカーソルが移動する。カーソルでないカーソルの移動に、効果的に使用される。Penを押しつけ続けると、キーボードと同様のリピート機能が働き、連続的にキーコードが送出される。リピート機能の時間間隔（繰り返し開始までの時間、繰り返す周期）は、キーボードのリピート機能に近く設定するが、必要に応じて変更もできる。

〔Hモード〕Xモードのバリエーション図13（c）に示すように画面の分割方法が異なるが、基本的機能はXモードに準じている。このモードでは、画面をH型にA、B、C、Dに4分割し、Penでポイントした位置に応じて、上／下／左／右のいずれかにカーソルが移動する。例えば、画面の右側をポイントすると、右側にカーソルが移動する。カーソルでないカーソルの移動に、効果的に使用される。Penを押し付け続けると、キーボードと同様のリピート機能が働き、連続的にキーコードが送出される。リピート機能の時間間隔（繰り返し開始までの時間、繰り返す周期）は、キーボードのリピート機能に近く設定するが、必要に応じて変更できる。ここでPモード、Xモードの動きを具体的に示す。図14はPモードで正常に動作しない状況を示す図である。

（a）はアプリケーションがファンクションキー等でサブメニューを表示し、反転カーソルで「保存」が選択されている。Pen 6で移動先「新規」をポイントすると、Pモードではカーソル70に対するポイント位置の

差分を出力するので、少なくとも2個以上の「下向き矢印」キーコードがアプリケーションに送られ、（b）のように反転カーソルは意図しない「取消」に移動してしまう。このような場合は前述のように自動またはマニュアルでXモードに切り替える。図15はXモードに切り替わった場合を示す図である。（a）でPen 6はXモードのD領域をポイントし、これに対応して「下向き矢印」のキーコードがアプリケーションに送られる。

（b）で「下向き矢印」を受け取ったアプリケーションは反転カーソルを「新規」に移動し、希望の動作を実行させることができる。図16はPモードからXモードに自動的に切り替わる際の処理を示す図である。PモードでPen 6が「新規」をポイントすると反転カーソルは（1）のように「取り消し」に移動してしまう。そこで（2）のように（1）の動作を一旦キャンセルして反転カーソルを「保存」に戻し、この時点でXモードに一時的に移行しPen 6がD領域をポイントすると（3）のように「新規」に移動する。

〔ファンクションガイドモード詳細〕未対応アプリケーションにおいて、画面最下行にファンクションガイドが表示されているものとして、その部分をポインティングデバイス（Pen／マウス等）で操作することにより、ファンクションキー入力や、代表的なキー入力が簡単に行なえるようにするものである。またスライドインやスライドアウトのように初期ポイント座標と最終ポイント座標のずれを利用して、より多くの情報を持たせる例外ファンクション機能もサポートする。上記スライドインとは目的のフィールドの外側でPenを押しつけて、そのままPenを離さずにスライドさせて目的のフィールド内に移動する動作であり、スライドインする方向には、意味がある場合がある。スライドアウトとはPenを押しつけたままスライドして、特定のフィールドから抜け出る動作であり、スライドアウトする方向には、意味がある場合がある。以下に図17を用いて動作を説明する。

（1） 通常のファンクションキーの入力を行ないたい場合は、76ないし85で示すキーブロック（K1,K2,...,KA）の中の1つをPenでポイントする（図18（a））。Penを押しつけている間、*で示すLB（レフトブランク）74；CB（センターブランク）89；RB（ライトブランク）90の位置（Penの位置、（利き腕にかかわらず見えるようにするため、3箇所と同じ表示を行なう；以下同様）にファンクションキーの名称を表示して、入力しようとするファンクションキーの確認ができるようにする。

（2） SHIFT+ファンクションキーの入力を行ないたい場合は、図18（b）のように、目的のキーブロックに左側からスライドインしてPenを持ち上げることによって入力できる。入力されるファンクションキーは、確認のため、*の位置に Sf・1 のように表示さ

れる。例えば、CB89でPenを押しつけ、K6__81までスライドさせて、そこでPenを持ち上げると、SHIFT+f・6が入力できる。この場合、Penを最初に押しつける位置は、K5__80または更に左側でもよい。

(3) CTRL+ファンクションキーの入力を行ないたい場合は、図18(c)のように、目的のキーブロックより右側からスライドインしてPenを持ち上げることによって入力できる。入力されるファンクションキーは、確認のため、*の位置にCf・1のように表示される。例えば、K7__82でPenを押しつけ、K6__81までスライドさせて、そこでPenを持ち上げると、CTRL+f・6が入力できる。この場合、Penを最初に押しつける位置は、K8__83または更に右側でもよい。

(4) LB74あるいはRB90をポイントすると、カーソルキーを含むキーエミュレーションのシフト状態を変更できる。シフト状態は、間隙をポイントするたびに、変更される*の位置に区別できるように表示される。(このシフト状態を「Penkeyのデフォルトのシフト状態」と定義する。SHIFT(LB74による)とCTRL(RB90による)とは、独立して変更できる。指定されたシフト状態は、固定に設定されない限り、何らかのキーをそのシフト状態で出力すると、通常のシフト状態に戻る。この操作に続いて、(1)の操作を行なうと、(2)または(3)の操作と同等の効果がある。但し、CTRL+SHIFT+ファンクションキーの入力は、(4)から(1)の操作以外では不可能である。

(5) キーブロックから上方へスライドアウトしてPenを持ち上げる動作に、よく使用するキー入力を割り当てる。図19にこれを示す。(a)はキーブロック内から垂直上方にスライドアウトする。すると表で示されるように、K1には「BS」、K2には「INS」というように各機能を割り当てる。また同じK1であっても、(b)のように左からスライドインし、上方にスライドアウトした場合は「SHIFT+BS」、さらに(c)のように右からスライドインし、上方にスライドアウトした場合は「CTRL+BS」を割り当てる。この割り当ては、ユーザの指定により変更することも可能である。

(6) CB89の右部分をポイントすると、カーソル移動モードをPモードとXモードとで交互に変更して、*にそのモードを表示する。このモードはカーソル移動モードで動作中は常に*に表示される。

P : Pモード X : Xモード (H : Hモード)

カーソル移動機能のキーリピートの間隔の変更を、例えば、下記の操作に割り当てることにより、Pen操作で動的に変更できるようになる。

a) 繰り返し間隔の変更

×2 = CB89に左からスライドインしてPenを持ち上げる

÷2 = CB89に右からスライドインしてPenを持ち上げる

b) 繰り返し開始までの間隔の変更

×2 = CB89に左からスライドインしてから上方にスライドアウトしてPenを持ち上げる

÷2 = CB89に右からスライドインしてから上方にスライドアウトしてPenを持ち上げる

この操作を実現するには、本実施例では後述するWaitに関する変数(InitWait、ReptWait)を変更するだけで、極めて容易に実現できる。

(7) Penを押しつけてしまってから、操作を取り消す(ファンクションガイドモード動作を途中でキャンセルするには、一旦最初にポイントしたキーブロックからスライドアウトして(方向はどちらでもよい)、再びそのキーブロックにスライドインする。このとき、*の表示は、——とする。Penkey58の動作説明の前に、各フローで使用される主なモードフラグ、変数の定義を挙げておく。いずれも情報保持部60の格納され、RAM24上に配置される。

※Penkeyモードフラグ定義

・PKM_INSLM : 1=ファンクションガイドモード

・PKM_MOUDN : 1=Pen押下中

・PKM_INPRG : 1=Penkey動作中

・PKM_WAIT : 1=P、X、ファンクションガイドモード初期動作終了時

・PKM_TMPX : 1=一時的なXモード(Xモード自動可変)

・PKM_CPINI : 1=初期カーソル位置保存完了(Pモードで使用)

・PKM_XCSR : 1=ユーザー指定によるXモード指定

※ウェイト変数

・InitWait : 初期ウェイト

・ReptWait : 繰り返し中のウェイト。

※ウェイト定義

・INIT_WAIT : 初期ウェイト定義

・REPT_WAIT : 繰り返しウェイト定義

・MIN_WAIT : 最小ウェイト定義。

※その他の制御変数

・WaitCnt : ウェイトカウンタ

・PX : 現在のPen位置 X座標

・PY : 現在のPen位置 Y座標

・PX0 : 初期Pen位置 X座標

・CX : 現在のカーソル位置 X座標

・CY : 現在のカーソル位置 Y座標

・CX0 : 直前のカーソル位置 X座標

・CY0 : 直前のカーソル位置 Y座標

・DX : PX-CX

- ・DY : PY-CY
- ・PC0 : 初期キーコード
- ・PC1 : 例外キーコード
- ・PC : 最終的に出力するキーコード
- ・Sh : Penkeyシフト状態
- ・XS : Penkey特別シフト状態

尚、各フラグ、変数の使い方は以降のフローチャートの中で説明する。以後フローチャートによる説明に入るが、各フローに付けられた「S〇〇〇」で示す符号はフローを識別するためのみ使用しており、実行の順序を示すものではない。図5は本発明のPenkey58の全体の流れを示すフローチャートである。アプリケーションがキー入力を要求すると(S10)、ポインティングデバイスに対応したアプリケーションではなく、Penkeyの停止をユーザーが指示しない限り、Penkeyが動作を開始する(S20)。まず、Penでポイントしている位置データであるポイントデータ(PX, PY)を取得し(S30)、その時点でまだPenが押下中であるかを調べる(S40)。押下中であればPenで初めてポイントしたかを調べ(S50)、そうであればポイントデータが最下行であるかをPYの座標データから判定する。最下行であればファンクションガイドモードに設定し(S70)、ファンクションガイドモード動作に入る(S160)。S60で最下行でない場合は、カーソルキーモードに設定し(S80)、さらに移動モードがPモードかを判定し(S100)、「Y」であればPモード動作に入り(S110)、「N」であればXモード動作に入る(S120)。S50で初めてのポイントでない場合は、すでにファンクションガイドモードに設定されている可能性もあるので、カーソルキーモードかを確認し(S90)、そうであれば動作モードの判定に入り(S100)、違う場合はファンクションガイドモード動作に入る(S160)。S40でPenが押下中でないと判断されると、ファンクションガイドモード後処理(S170)、Xモード後処理(S180)、Pモード動作(S110)内の後処理部分、が終了したかを調べ、Penkeyを抜けてオリジナルのキー入力処理部52に制御を戻す。この後キー入力処理部52はキーコード待ち行列54に保持されたPCをキーボード32から入力されたキーコードと同様にアプリケーションに渡す。S130で「N」であればカーソルキーモードかを判定し(S140)、「N」でファンクションガイドモード後処理(S170)に入り、「Y」で、動作モードがPモードであればPモード動作(S100)に入って内部の後処理ルーチンと実行し、そうでなければXモード後処理(S180)を行う。Pモード動作(S110)、Xモード動作(S120)終了後、PCがなければS210に入り、あればキーコード出力リピータ処理(S200)を経た後S210に入る。ファンクションガイドモード後処理(S170)、Xモード

ド後処理(S180)後も同様にS210に入り、PCを出力してPenkeyを抜ける。Pモード動作(S110)の説明に入る前に図12を利用して、Pモードでカーソル70を移動している際の表示画面を説明する。P1の位置にあるカーソル70は、Pen6で移動先P2をポイントすると(a)、カーソル移動キーで操作したと同等にP2に移動する(b)。この時、Pen6を押し付け続けると、カーソル70は操作者に見易いようにゆっくりと移動し、すぐにPen6を放すと、放した後高速に移動するように制御される。図6は、Pモード動作を詳細に説明するためのフローチャートである。S110で動作に入り、まず現在のカーソル位置(CX, CY)を取得する(S300)。初めてのポイントかを確認し(S310)、「Y」であれば現在のPen位置(PX, PY)の対する現在のカーソル位置(CX, CY)のずれ量(DX, DY)を求め(S390)、PCを初期化する(S400)。次に垂直方向のずれDYを確認して(S410)、DY>0であればPCに「下向き矢印」に対応するキーコードを出力し(S420)、同様にDY<0であればPCに「上向き矢印」を出力して(S430)、S470で現在のカーソル位置(CX, CY)を直前のカーソル位置(CX0, CY0)として保存し、Pモード動作に抜ける。S410でDY=0の場合は、水平方向のずれ量DXを確認し(S440)、DX>0であればPCに「右向き矢印」を出力し(S450)、DX<0であれば「左向き矢印」を出力し、それぞれにS470に入る。またDX=0であれば、なにもせずにS470に入る。以上によりPCにセットしたキーコードは、S200のキーコード出力リピータ処理の段階でキーコード待ち行列54に出力される。S310で「N」の場合は、前回何等かのキーコードが送出されたはずであるから、カーソルの移動があったかを確認することになる。前回のキーコードをPCから呼び出し、「上向き矢印」または「下向き矢印」であれば(S320)、カーソルが上下に動いたかを確認し(S330)、動いていれば正常と判断してS390以降で次のキーコードを発生する。上下に動いていなければ次に左右のカーソルの動きを確認し(S340)、左右にも動いていなければカーソルが移動できない状態であると判断して、前回と逆の動作を設定し、PKM_TMPX77gを操作して一時的にXモードにする。「上向き矢印」「下向き矢印」に対しカーソルが動かない場合とは、例えば図14のようにサブメニューが出た状態等である。この時、自動的にXモードに切り替わり、入力終了後、再びPモードに復帰する。S340でカーソルが左右に動いていれば、意図と違う動きをしていると判断し、Penkeyを停止し(S380)、S470に入る。S320で前回「上向き矢印」「下向き矢印」でなければS360でオーバーランを判定する。オーバーランとは(PX, PY)に対し、カーソルが行き過ぎた場合、ま

た、〔Pモード詳細〕の4)で記載した3項目のように、まったく意図しない場所にカーソルが飛んだ場合もオーバーランで処理する。オーバーランと判定されると、前回と逆の動作を行ってPenkeyを停止し、S470に入る。S360でオーバーランでなければ左右のカーソルの動きを確認し、移動していれば正常と判断し、S390に入る。移動がなければ、目標に達したと判断しS380でPenkeyを停止し、S470に入る。以上がPモードの動作である。次に図7のフローチャートを用いてXモード動作を説明する。図5のS100でPモードでなければ、S120でXモードに入る。S600でPCをクリアし、ポイント位置(PX, PY)が図13(b)の領域Aであれば(S610)、「左向き矢印」に対応するキーコードをPCに出力する。以後同様にポイント位置が領域Bであれば「右向き矢印」、領域Cであれば「上向き矢印」、領域Dであれば「下向き矢印」のキーコードをPCに出力し、Xモードを抜ける。次に図8のフローチャートを用いて、キーコード出力リピート処理を説明する。まず、ウェイトの必要性であるが、〔Pモード詳細〕の項目でも述べたように、カーソル移動の実処理に対しPenkeyの処理が早すぎると、カーソルの移動が検出されず、Pモードの動作がすぐに停止してしまうことを避けるためである。PモードあるいはXモードでPCがセットされている、S200のキーコード出力リピート処理に入る。S1000でPenが押下中か判定し、「N」であればWaitCnt(カウントダウンを行うウェイトカウンタ)を、初期値として経験的に設定している最小ウェイト:MINWaitと比較し、WaitCntが大きければWaitCntにMINWaitの値を設定する。これは本処理を無意味に遅くしないためである。この後S1030に入る。S1000で「Y」の場合、S1010で「N」の場合はそのままS1030に入る。ここで初めてのポイントでなければ、WaitCnt=0かを判定し(S1040)、「N」であればWaitCntを1カウントダウンし(S1110)、何もせずに終了する。S1030で「Y」であればS1040をスキップして、PKShiftをセットしてPenkeyのシフト状態(ファンクションガイドの操作により予め設定されたシフト状態)にし(S1050)、PモードあるいはXモード動作で設定されたキーコード:PCをBIOSキーバッファであるキーコード待ち行列54に出力する(S1060)。これが初めてのキー出力であれば(S1070)、初期ウェイト値(〔Pモード詳細〕等で述べた繰り返し開始時間のこと):InitWaitをWaitCntに設定し、シフト状態をノーマルに戻して(S1100)処理を抜ける。S1070で「N」であれば繰り返し中のウェイト(〔Pモード詳細〕等で述べた繰り返し開始時間のこと):ReptWaitをWaitCntに設定し、S1100を経て、キーコード出

カリピート処理を終了する。次に図9のフローチャートを用いてXモード後処理を説明する。S180でXモード後処理に入り、PKM_TMPXフラグを見て一時的なXモードかを判定し(S1400)、「Y」であればPモードに戻して処理を終了し、「N」であれば現在のモードのままXモード後処理を終了する。次に図10のフローチャートを用いて、ファンクションガイドモード動作の説明を行う。S160でファンクションガイドモード動作に入り、ポイント位置が最下行かを判定し(S800)、最下行であって、図17のLB74の領域であれば(S810)、「SHIFT」によるシフト状態を変更し(S820)、RB90の領域であれば(S830)、「CTRL」によるシフト状態を変更し(S840)、CB89の領域であれば(S850)、モード(PorX)を変更し合わせて表示も変える。S820、S840、S860が終了するとファンクションガイドモード動作を抜ける。S850で「N」と判断されると、Penでポイントした点のX座標:PXから、これに対応したファンクションコード:fk(PX)を決める。これは各キーブロックと出力するキーコードを対応させたテーブルをPenkeyに持っていればよい。次にS890で初めてのポイントかを確認し、「Y」であれば、現在のPXおよびPCを、PX0、PC0として保存し、処理を終了する。つまり、図18や図19で示すような、例外ファンクション機能をサポートするため、初期値と最終的なポイントで出力キーコードが決定されることになる。S890の「N」の分岐以降を述べる前に、S870を説明する。S800で最下行でない判定されると、S870で特別なキーコードの設定を行う。つまり、PX、PX0に対応するキーコード:xk(PX、PX0)を定めるテーブルをPenkeyに設け、決定されたキーコードを例外キーコードとしてPC1に保存する。例えば図19の表の「BS」、「INS」、「DEL」等は通常ファンクションキーには割り当てられていないが例外キーコードとして出力可能である。またPYと新たにPY0を検出すればさらに多くの機能割り付けが可能になる。またPX0とPXの組み合わせを特別シフト状態と定義することもできる。つまりxks(PX、PX0)=XSとしてXSフラグを特別シフト状態に設定する。特別シフト状態とは、図19の表でK6ないしKAのように、「スペース」のシフト状態が「TAB」、「Y」のシフト状態が「N」というように、独自のテーブルを作ることである。S870を終えると、S940で初期キーコード:P0がキャンセルされているかを確認し(S940)、「Y」で処理を抜け、「N」であればS950発生した例外キーコードをPC1とPC0にセットする。S890で「N」であれば、PC0とPCが一致するかを確認し(S910)、一致していなければS940に進み、一致していればS920でPC1を見て、例外発生かを確認する。「Y」であれば処

理を抜け、「N」であればPC0をキャンセルしてから処理を抜ける。最後に図11のフローチャートを用いて、ファンクションガイドモード後処理のルーチンを説明する。S180でファンクションガイドモード後処理に入り、S1600で***PCが有効かを判定する。ここでPC0はキャンセル動作があった場合、初期ポイントが無効だった場合（例えばK1_76とK2_77の隙間のようなポイントを無効とする）にNULLになる。無効であれば何もせずに処理を終了する。有効であればPC1を見て例外有かを識別し（S1610）、例外でなければ、S1620で、Penkeyのデフォルト（初期設定状態）のシフト状態に変え、S1670でPCをキーコード待ち行列54に出力し、S1680を通過して、処理を抜ける。S1610で例外キーコードがあれば、S1630で特別シフト指定であるかを判定し、そうであればシフト状態をXSで指定された状態とし（S1640）、S1680でPCを出力し、S1680でシフト状態を通常に戻して、処理を抜ける。S1630で特別シフト状態でないと判断すると、PXとPX0を比較し（S1640）、 $PX > PX0$ であれば「SHIFT」（S1650）、 $PX < PX0$ であれば「CTRL」（S1660）にシフト状態を設定してPCをキーコード待ち行列に送り出し（S1670）、S1680を経て、処理を終了する。以上、Pen入力による情報処理装置を例として述べてきたが、本発明はこれに限定されるものではない。マウス、トラックボール等、座標を特定する手段を有するデバイスであれば、すべて同様に実施可能である。またファンクションモードガイドによる入力も、本実施例では座標を2点以上検出して識別しているが、ポインティングデバイスの移動速度、加速度、ポイントする時間間隔、ポイントする強さ等、さまざまな要因で識別可能である。

【発明の効果】本発明の情報処理装置はポインティングデバイス等による位置情報を保持する位置指定手段と、前記位置指定手段により出力された情報を、キーコード情報に変換する変換手段と、前記キーコード情報を保持するコード処理手段と、前記コード処理手段からの情報で制御される表示手段とを有するので、多種の位置情報をキーコード情報、シフト情報等の状態設定情報に変換して利用することができる。

【図面の簡単な説明】

【図1】本発明の基本構成を例示するブロック図である。

【図2】本発明の一実施例である情報処理装置16の外観図である。

【図3】本発明の一実施例である情報処理装置16のハードウェア構成図である。

【図4】本発明の一実施例である情報処理装置16の全体の処理の流れを示す図である。

【図5】Penkey58の全体の流れを示すフローチャートである。

ャートである。

【図6】Pモード動作のフローチャートである。

【図7】Xモード動作のフローチャートである。

【図8】キーコード出力リピート処理のフローチャートである。

【図9】Xモード後処理のフローチャートである。

【図10】ファンクションガイドモード動作のフローチャートである。

【図11】ファンクションガイドモード後処理のフローチャートである。

【図12】Pモードでカーソル70を移動している際の表示画面を説明する図である。

【図13】XモードおよびHモードの操作画面を説明する図である。

【図14】Pモードで正常に動作しない状況の表示画面を示す図である。

【図15】Pモードで正常に動作しない状況において、Xモードに切り替わった場合の表示画面を示す図である。

【図16】PモードからXモードに自動的に切り替わる際の反転カーソルの処理を示す図である。

【図17】ファンクションガイドモード動作の際の表示画面を示す図である。

【図18】スライドイン、スライドアウトによりキーブロックに「SHIFT+F・?」、「CTRL+F・?」の意味を持たせることを説明する図である。

【図19】Penの操作によりキーブロックに多くの機能を割り付けることを示す図である。

【図20】本発明の従来例であり、ポインティングデバイスに対応したアプリケーションプログラムのカーソル200の移動動作を示す図である。

【図21】本発明の従来例であり、ポインティングデバイスに未対応のアプリケーションプログラムに対し、ソフトウェアキーボード204を表示して、カーソル200の移動を行っていることを示す図である。

【符号の説明】

2：Pen入力タブレット

4：表示部

6：Pen

8：バッテリー

10：カバー

12：カバー

14：カバー

16：情報処理装置

20：CPU

22：ROM

24：RAM

26：タブレットI/F

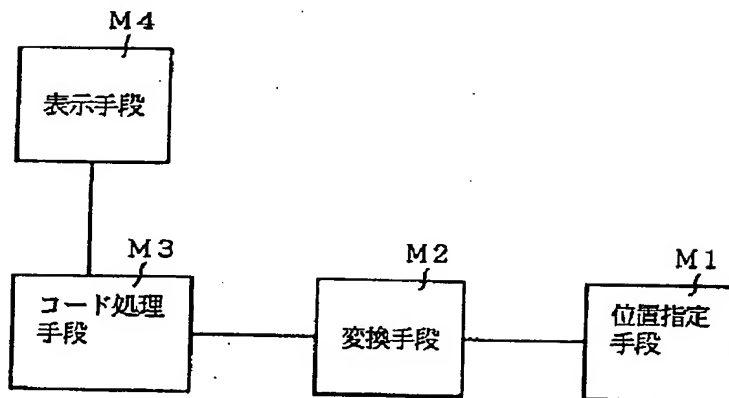
28：表示制御部

30：キーボード

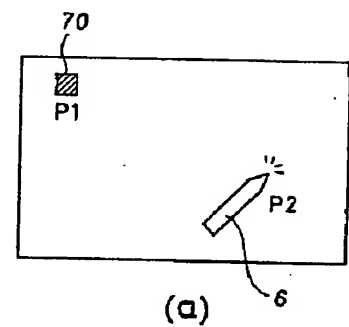
32 : キーボード
 34 : 外部記憶 I/F
 36 : 外部記憶装置
 38 : タイマ
 40 : システムバス
 50 : 未対応アプリケーション
 52 : キー入力処理部
 54 : キーコード待ち行列
 56 : Pen 入力処理部
 58 : Pen key
 60 : 情報保持部
 70 : カーソル
 72 : カーソル移動モードブロック
 74 : LB (レフトブランク)
 76 : K1
 77 : K2
 78 : K3

79 : K4
 80 : K5
 81 : K6
 82 : K7
 83 : K8
 84 : K9
 85 : KA
 89 : CB (センターブランク)
 90 : RB (ライトブランク)
 10 200 : カーソル 200
 202 : ポイント指示カーソル
 204 : ソフトウェアキーボード
 M1 : 位置指定手段
 M2 : 変換手段
 M3 : コード処理手段
 M4 : 表示手段

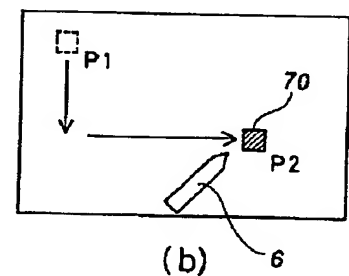
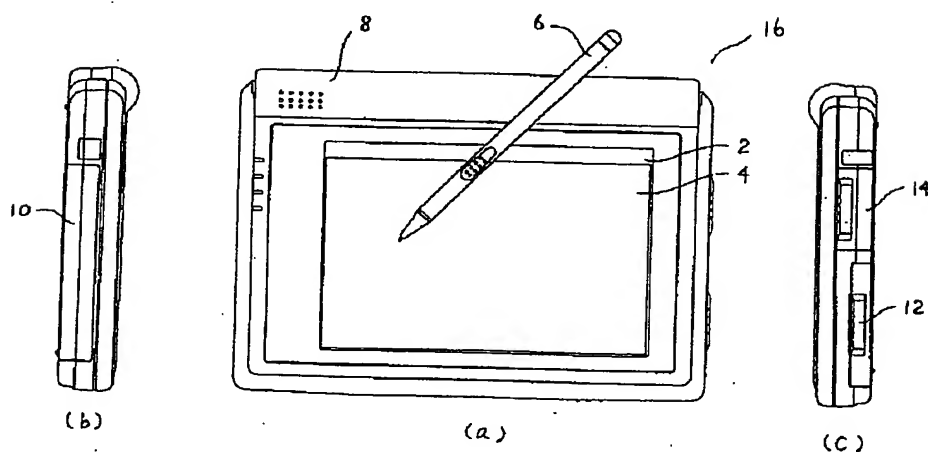
【図1】



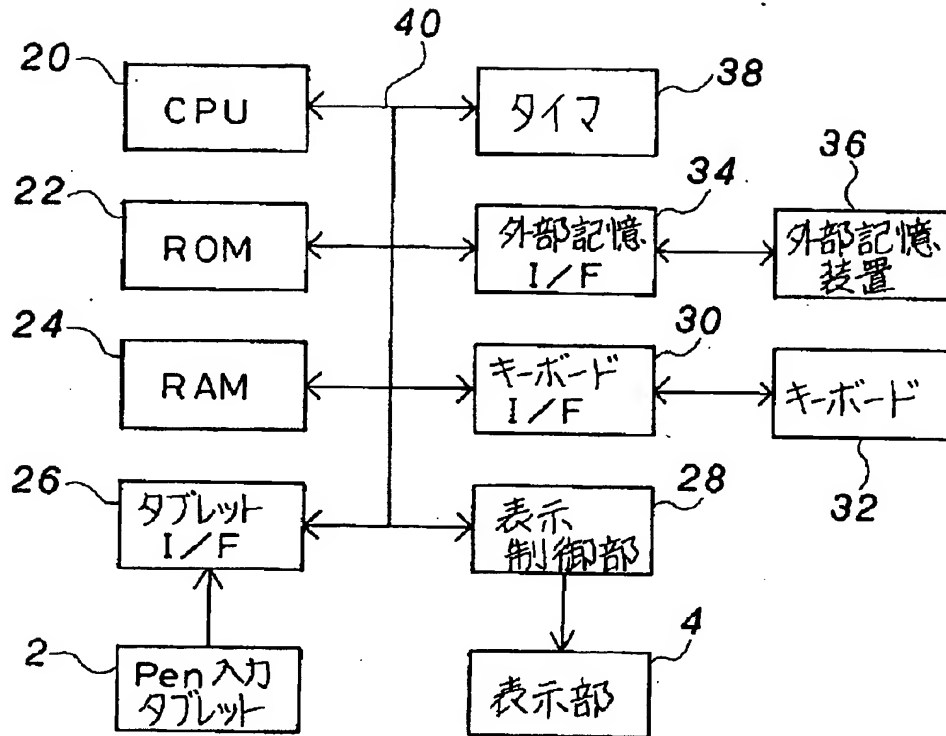
【図12】



【図2】

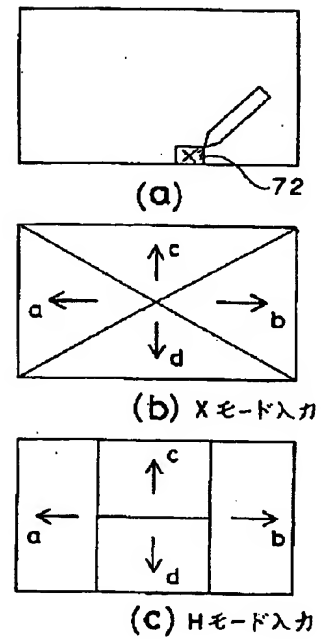


【図3】

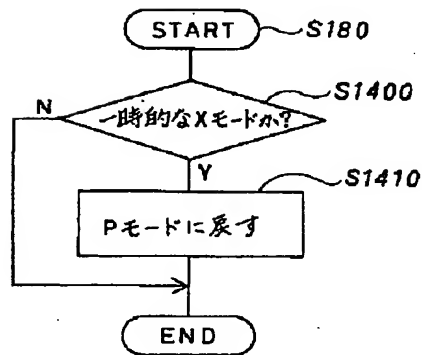


本発明のハードウェア構成図

【図13】

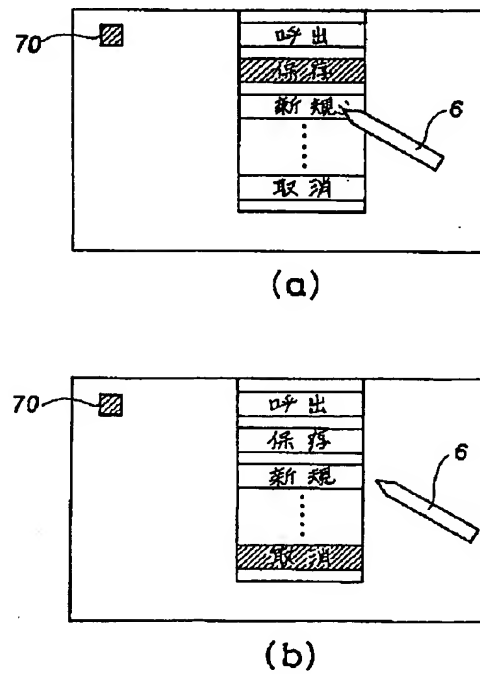


【図9】

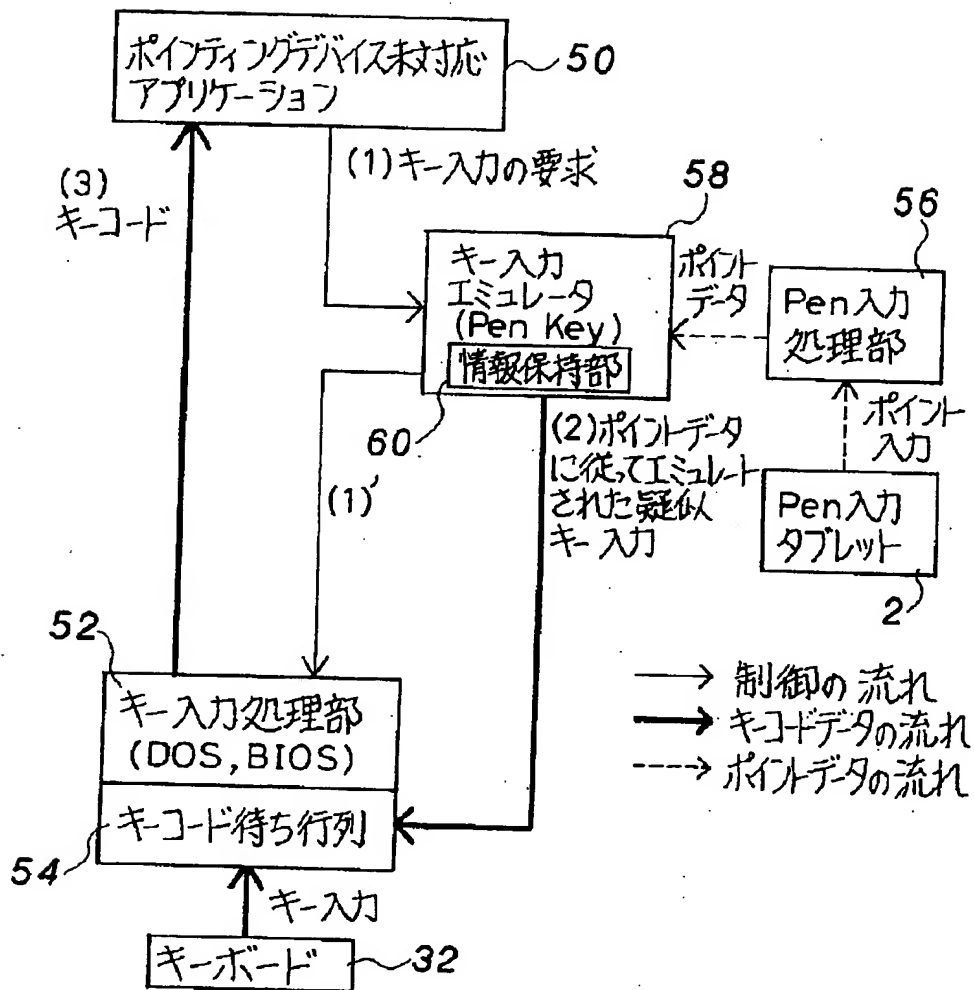


Xモード後処理

【図14】

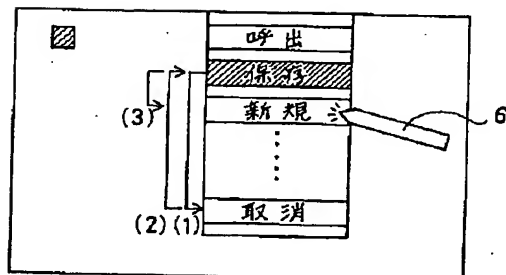


【図4】

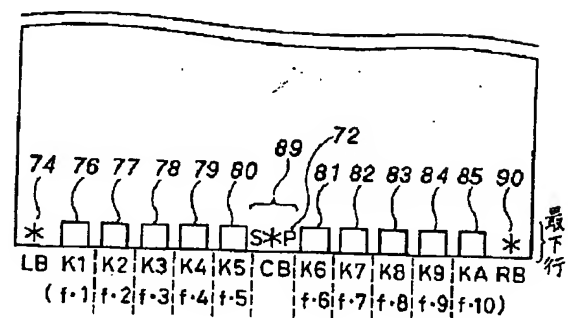


本発明の全体の流れを示す図

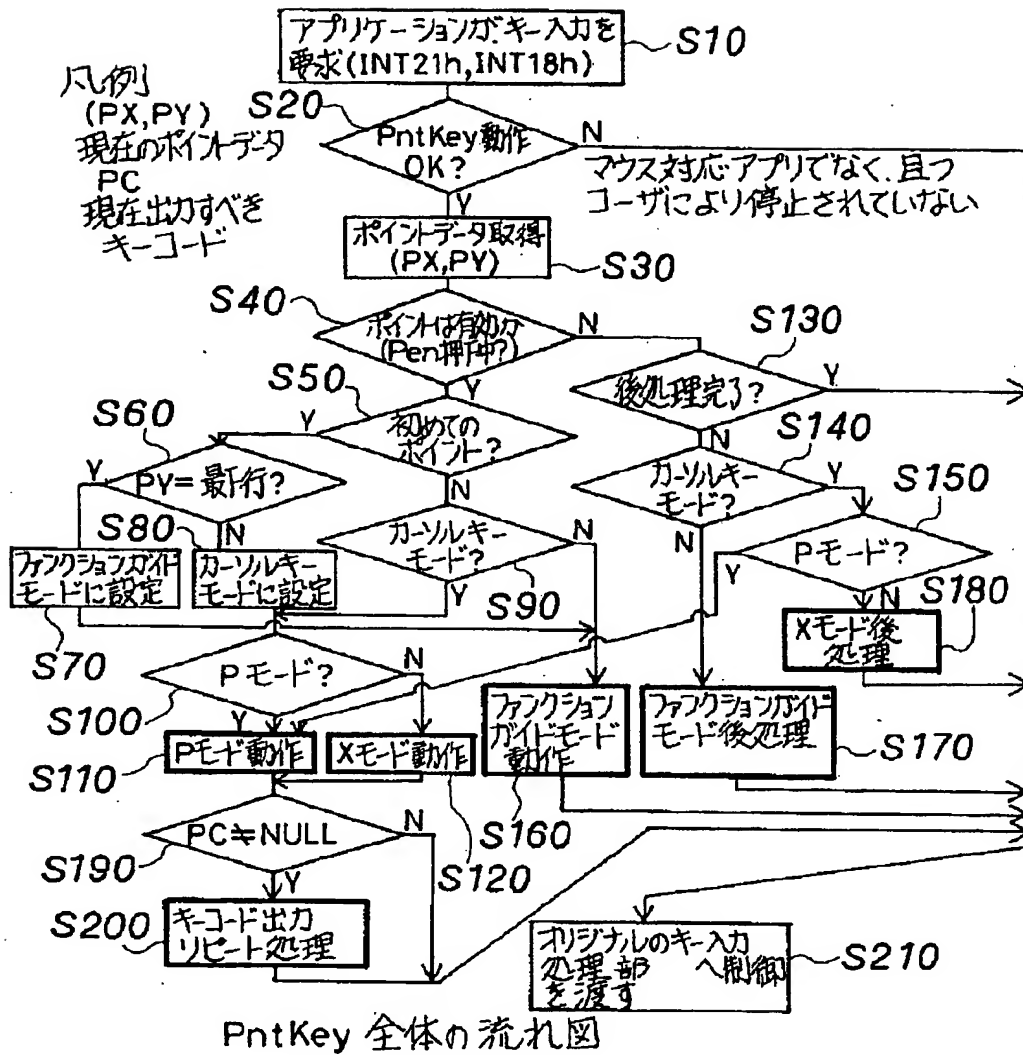
【図16】



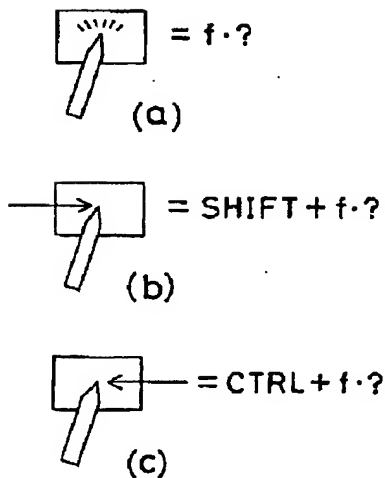
【図17】



【図5】



【図18】

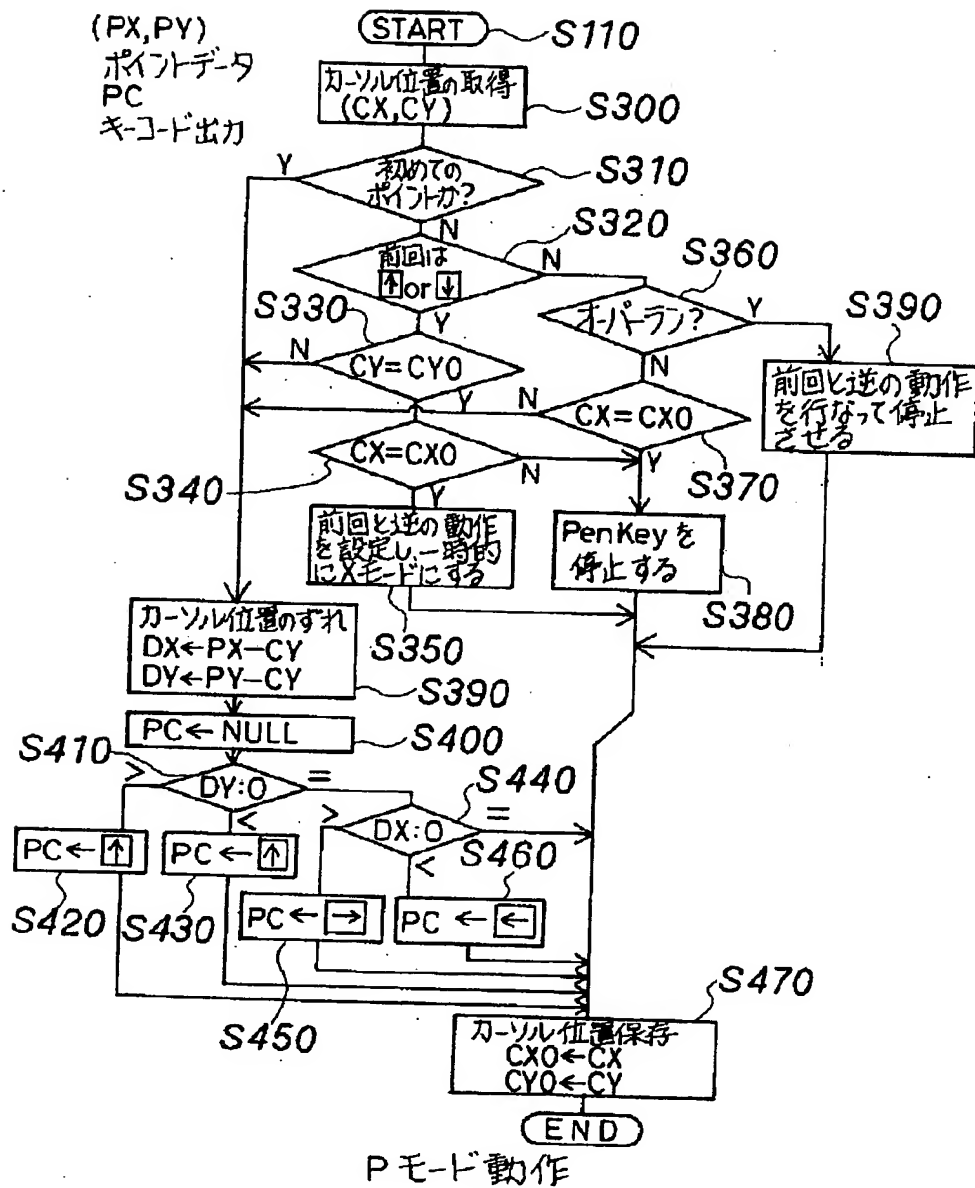


【図19】

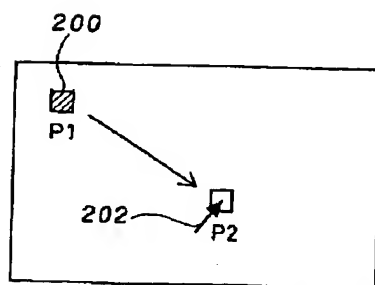
K1	BS	SHIFT+BS	CTRL+BS
K2	INS	SHIFT+INS	CTRL+INS
K3	DEL	SHIFT+DEL	CTRL+DEL
K4	HOME CLR	SHIFT+HOME	CTRL+HOME
K5	HELP	SHIFT+HELP	CTRL+HELP
K6	スペース	TAB	ROLL UP
K7	Y	N	ROLL DOWN
K8	A	R	F
K9	ESC	SHIFT+ESC	CTRL+ESC
KA	リターフ	SHIFT+リターフ	CTRL+リターフ

↑
↑
↑ Penの操作

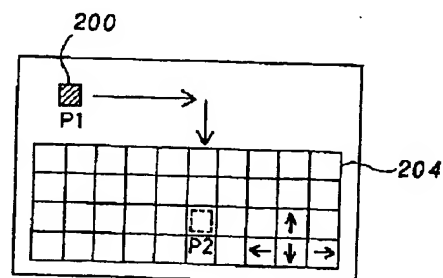
【図6】



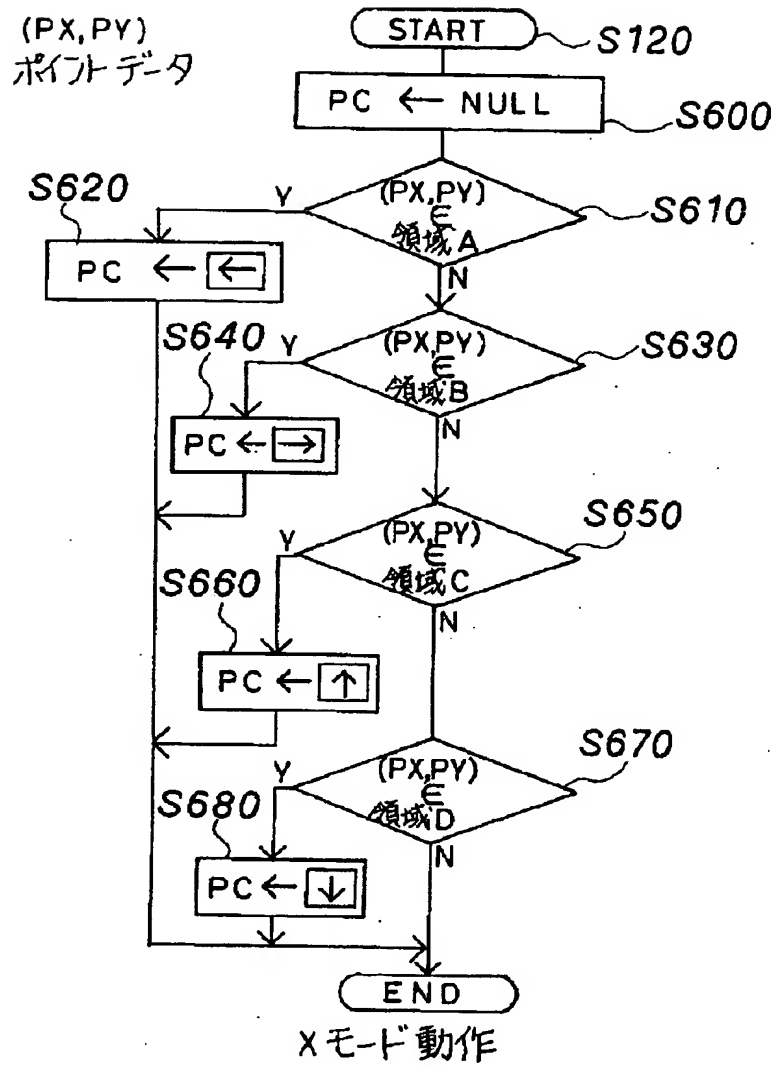
【図20】



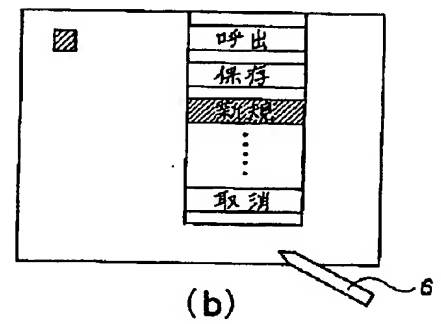
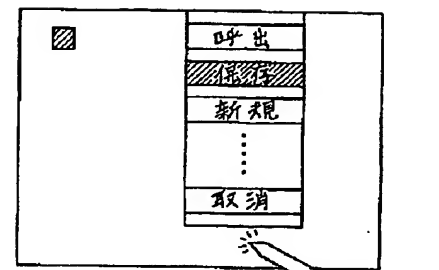
【図21】



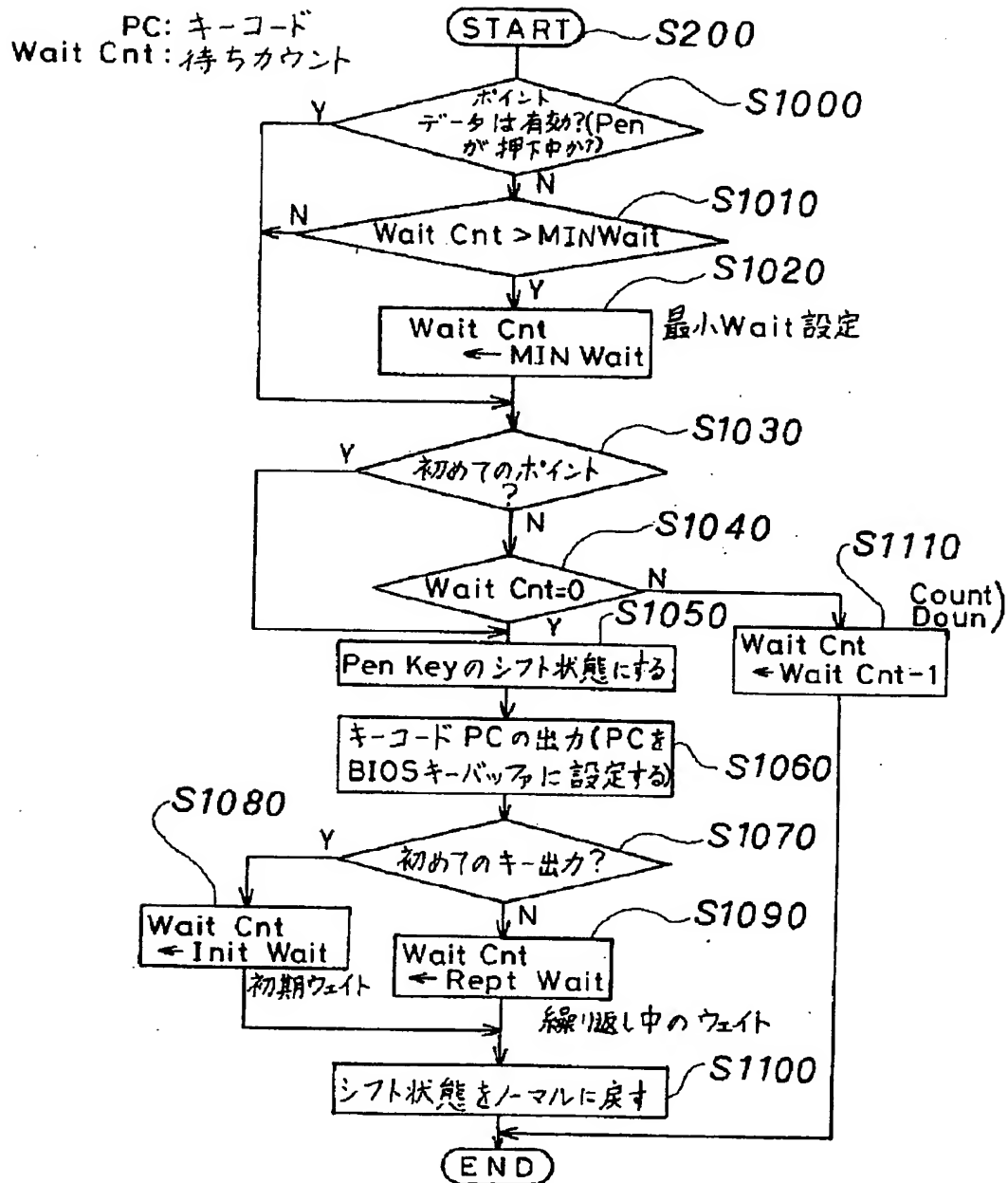
【図7】



【図15】

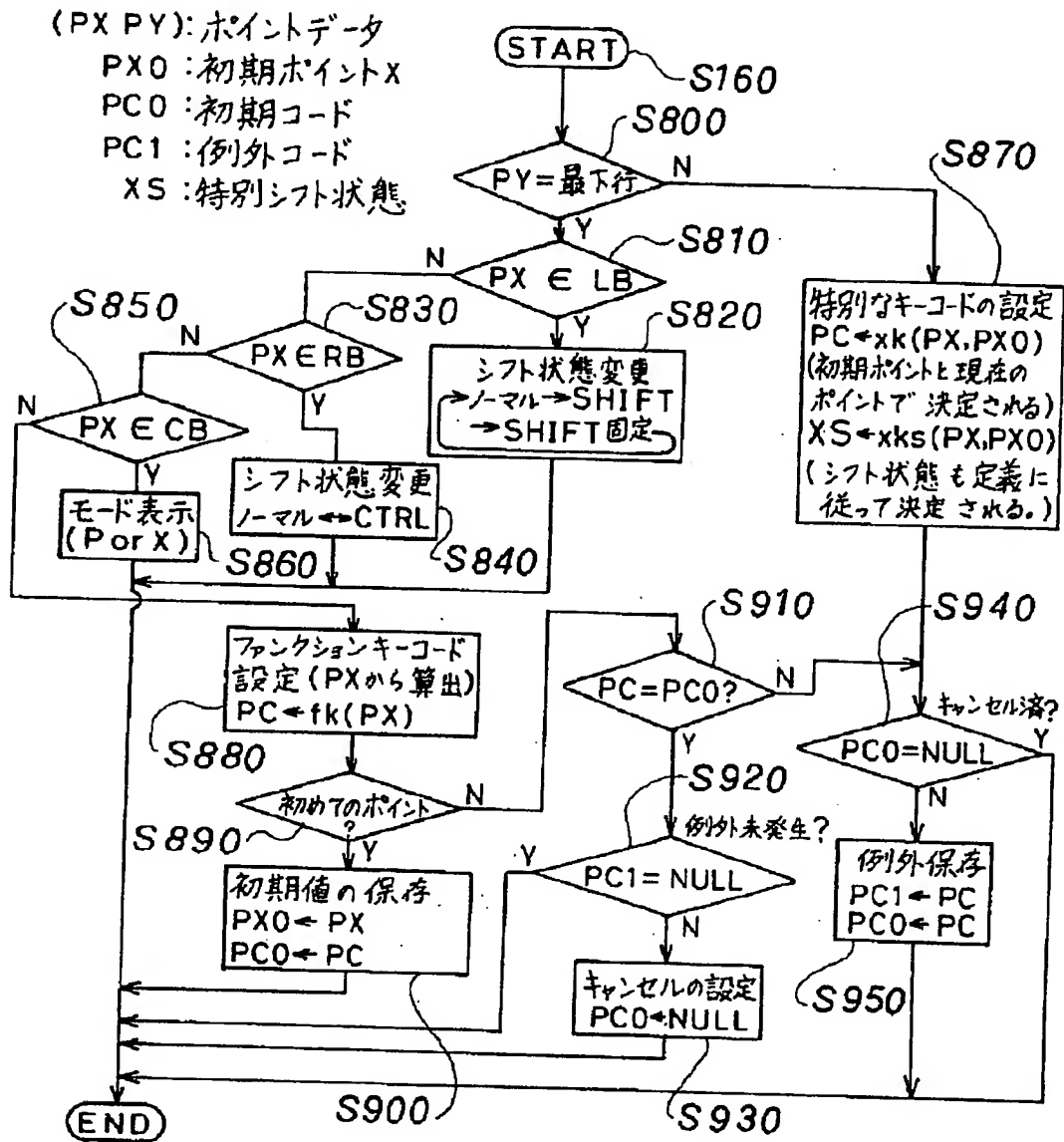


【図8】



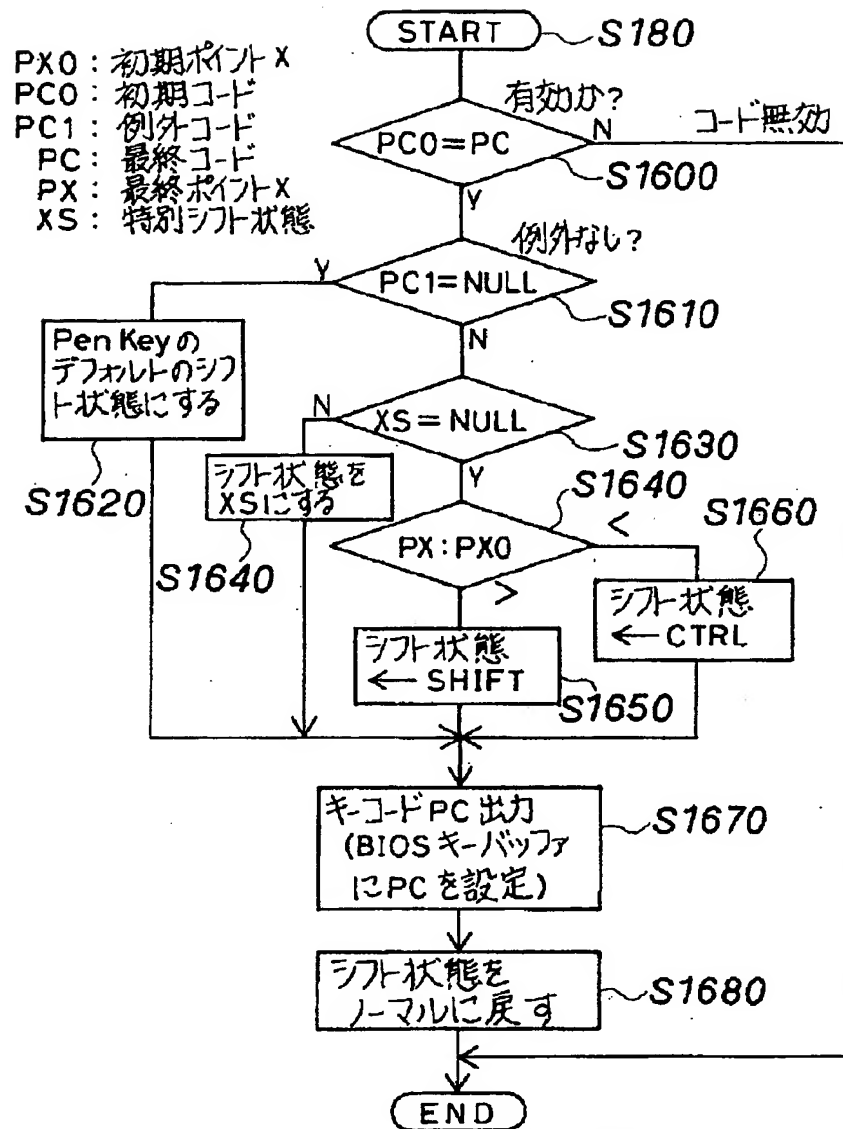
キーコード 出力リポート処理

【図10】



ファンクションガイドモード動作

【図11】



ファンクションガイドモード後処理